

SESSION

I. LATAR BELAKANG

Sebuah web server tidak dapat mengenali siapa saja yang sedang mengakses dan apa yang dilakukan oleh *user* karena protokol HTTP bersifat *stateless*. Sebuah protokol dinyatakan *stateless* karena tidak menyimpan informasi mengenai interaksi yang terjadi sebelumnya, sehingga setiap interaksi yang terjadi dianggap sebagai interaksi yang independen dan tidak terkait satu sama lain.

Ketika *user* mengakses suatu halaman website, misalnya index, dan berpindah ke halaman lain di website yang sama, *webserver* akan menganggap setiap interaksi yang bersifat *request-response* tersebut sebagai interaksi tunggal. Tidak ada kaitannya antara satu halaman dengan halaman yang lain.

Agar *webserver* dapat 'mengingat' informasi pengakses, PHP menyediakan sebuah mekanisme yang disebut *session*. Dengan adanya *session*, setiap pengakses diberi pengenal (ID) yang akan digunakan selama mengakses sebuah website. Pengenal tersebut dapat didefinisikan oleh pembuat programer. Informasi pada *session* disimpan di server. Oleh sebab itu, *session* akan hilang secara otomatis jika user menutup browser atau meninggalkan (ditutup/ke luar) website tersebut. User akan mendapatkan pengenal yang baru ketika mengakses kembali halaman website yang sama.

Session memiliki fungsi seperti tiket di bioskop atau di tempat-tempat yang membutuhkan tiket. Fungsi tiket untuk memastikan bahwa seseorang berhak untuk masuk ke dalam bioskop. Petugas bioskop hanya dapat mengizinkan seseorang untuk masuk ke bioskop, apabila dapat menunjukkan tiket yang telah dibelinya. Apabila di tengah pertunjukan seorang penonton ke luar sebentar, misal untuk membeli makanan atau ke toilet, maka tiket yang sama harus ditunjukkan kepada petugas penjaga agar dapat masuk kembali ke bioskop. Jika ternyata tiket hilang, maka petugas penjaga berhak melarang penonton tersebut untuk masuk kembali.

Pemanfaatan *session* sangat terasa ketika menggunakan aplikasi berbasis web seperti *e-commerce*, media sosial seperti Twitter dan Facebook, atau blog.

II. MEMULAI SESSION

Session dimulai dengan menggunakan function `session_start()`, perhatikan kode di bawah ini:

```
<?php
session_start();
?>
<html>
  <head>
    <title>Session</title>
  </head>
  <body>
    <h1>Latihan Session</h1>
  </body>
</html>
```

Function `session_start()` harus dipanggil sebelum aktivitas lain dilakukan. Jika dalam sebuah halaman website, pastikan `session_start()` berada di atas tag `html`. Selain itu, `session_start()` harus dipanggil di seluruh halaman yang akan menggunakan informasi yang tersimpan pada session.

III. MENYIMPAN INFORMASI KE DALAM SESSION

PHP menyediakan variabel superglobal `$_SESSION` untuk menyimpan informasi ke dalam session. `$_SESSION` merupakan variabel dengan tipe data array, seperti `$_POST`, `$_GET` atau `$_SERVER`. Perhatikan contoh di bawah ini:

```
<?php
session_start();
//menyimpan informasi ke dalam session
$_SESSION['user'] = "Taufik Hidayat";
?>
<html>
  <head>
    <title>Session</title>
  </head>
```

IV. MENGAMBIL INFORMASI DARI SESSION

`$_SESSION` merupakan sebuah array, oleh sebab itu untuk dapat mengambil nilai dari sebuah variabel `$_SESSION`, dilakukan dengan cara yang sama mengambil nilai dari sebuah array. Contoh:

```
<?php
session_start();
//menyimpan informasi ke dalam session
$_SESSION['user'] = "Taufik Hidayat";

//menampilkan nilai session
echo "Nama :".$_SESSION['user'];

//menyimpan nilai session ke variabel
$nama_user = $_SESSION['user'];
echo $nama_user;

?>
<html>
  <head>
    <title>Session</title>
  </head>
```

V. SESSION FUNCTIONS

Berikut ini beberapa function yang umum digunakan

- `session_start()`, untuk memulai session
- `session_id()`, untuk menset atau mendapatkan id *session*. Contoh:

```
$sid = session_id();
echo "Session ID: ".$sid;
```

- `session_regenerate_id()`, untuk meng-update session id.

```
session_start();  
$sid_lama = session_id();  
session_regenerate_id();  
$sid_baru = session_id();  
  
echo "Session ID Lama: ".$sid_lama;  
echo "<br>Session ID Baru: ".$sid_baru;
```

- `session_name()`, untuk menset dan mendapatkan nama session.

```
session_start();  
echo session_name();  
session_name( 'APLIKASI' );  
echo session_name();
```

- `session_destroy()`, untuk menghapus session

```
//menyimpan informasi ke dalam session  
$_SESSION['user'] = "Taufik Hidayat";  
  
//menampilkan nilai session  
echo "Nama :".$_SESSION['user'];  
  
//menyimpan nilai session ke variabel  
$nama_user = $_SESSION['user'];  
echo $nama_user;  
  
session_destroy();  
echo "Nama :".$_SESSION['user'];
```

LATIHAN

1. Buat sebuah folder dengan nama session pada folder server
2. Buat file **index.php**

```
<?php session_start(); ?>
<html>
  <head>
    <title>Session</title>
  </head>
  <body>
    <?php
    if ( $_SESSION['login'] === 1 ){
      echo '<h1>Selamat Datang ' . $_SESSION['username'] . '</h1>';
      echo '<a href="logout.php">Log Out</a>';
    }
    else {
      require_once( 'form.php' );
    }
    ?>
    <a href="halaman2.php">Halaman 2</a>
  </body>
</html>
```

3. Buat file form.php

```
<form method="post" action="login.php">
  <table>
    <tr>
      <td>Username:</td>
      <td><input type="text" size="20" name="username"></td>
    </tr>
    <tr>
      <td>Password:</td>
      <td><input type="password" size="20" name="password"></td>
    </tr>
    <tr>
      <td colspan="2"><input type="submit" value="Log In"></td>
    </tr>
  </table>
</form>
```

4. Buat file **login.php** untuk menangani data login

```
<?php
session_start();

$user = "admin";
$pass = "123456";

if ( !$_POST ){
    die( "Maaf, Anda tidak bisa login" );
}

$username = trim($_POST['username']);
$password = trim($_POST['password']);

if ( ( $user == $username ) && ( $pass == $password ) ){
    $_SESSION['username'] = $username;
    $_SESSION['login'] = 1;
    echo "Selamat Datang Kembali: ".$username;
    echo "<br><a href='index.php'>Ke halaman Index</a>";
    echo " <a href='halaman2.php'>Ke Halaman 2</a>";
}
else {
    die( "Maaf, Anda tidak memiliki akses" );
}
?>
```

5. Buat file **halaman2.php**

```
<?php
session_start();
if ( $_SESSION['login'] != 1 ){
    die( "Anda tidak memiliki hak akses" );
}
?>
<html>
<head>
<title>Halaman 2</title>
</head>
<body>
<h1>Selamat Datang di Halaman 2: <?php echo $_SESSION['username']; ?></h1>
<a href="index.php">ke Halaman Index</a>
</body>
</html>
```

6. Terakhir, buat file **logout.php**

```
<?php  
session_start();  
session_destroy();  
  
//redirect ke halaman index.php  
header( 'location: index.php' );  
?>
```

REFERENSI

7. Sessions, <http://www.php.net/manual/en/features.sessions.php>
8. Session Handling, <http://www.php.net/manual/en/book.session.php>
9. PHP Sessions, http://www.w3schools.com/php/php_sessions.asp